

# Semidefinite Relaxations for Stochastic Optimal Control Policies

Matanya B. Horowitz, Joel W. Burdick

**Abstract**—Recent results in the study of the Hamilton Jacobi Bellman (HJB) equation have led to the discovery of a formulation of the value function as a linear Partial Differential Equation (PDE) for stochastic nonlinear systems with a mild constraint on their disturbances. This has yielded promising directions for research in the planning and control of nonlinear systems. This work proposes a new method obtaining approximate solutions to these linear stochastic optimal control (SOC) problems. A candidate polynomial with variable coefficients is proposed as the solution to the SOC problem. A Sum of Squares (SOS) relaxation is then taken to the partial differential constraints, leading to a hierarchy of semidefinite relaxations with improving sub-optimality gap. The resulting approximate solutions are shown to be guaranteed over- and under-approximations for the optimal value function.

## I. INTRODUCTION

As robots and autonomous systems are fielded in increasing complex situations, the ability to move safely in the presence of uncertain actuation and sensing, as well as dynamically changing and uncertain environments, becomes ever more important. Practically useful control and planning methods must also be rapidly computable, and should incorporate optimality criteria when possible.

Sampling based planners such as PRMs and RRTs [10] have become popular for such problem because they are adaptable to a variety of problems, and often have rapidly computable solutions in higher dimensional problems. However, these approaches typically rely on an abstraction of the state space that eliminates considerations such as stochasticity and dynamics. Secondary issues such as control effort and movement efficiency cannot also be readily incorporated in these main stream approaches.

Stochastic optimal control (SOC) provides an alternative framework, allowing for various important details of the problem to be directly incorporated into the motion planning formulation and solution. Many such SOC problems are discretized, resulting in Markov Decision Problems that can be solved through methods such as Value Iteration [1]. In robotic applications, such discretizations become prohibitively difficult to solve due to the curse of dimensionality associated with robotic systems of even moderate complexity.

Recently it has been discovered that the (typically nonlinear) Hamilton Jacobi Bellman (HJB) equation of optimal control may be transformed to a linear PDE given several mild assumptions [9], [25]. This approach might lead to significant computational gains, allowing for practical applications of SOC. To date, this class of linearly solvable

HJB problems has been solved through sampling methods suggested by the Feynman-Kac Lemma [20].

This paper presents a novel alternative method to solve such problems using polynomial optimization and semidefinite programming. Using *sum of squares* (SOS) techniques, [14], we construct an approximate value function that satisfies the linearly solvable HJB equation. This allows for optimal control problems, including those typically found in robotic motion planning, to be computed quickly, with globally optimal solutions. In contrast to dynamic programming approaches, no discretization is required, postponing the curse of dimensionality and eliminating a potential source of approximation error. Moreover, our formulation leads directly to gap theorems, or bounds, on the approximation error.

**Related Work.** The study of linearly solvable SOC problems has developed along two lines of investigation. One is that of Linear MDPs [25], in which an MDP may be solved as a linear set of equations given several assumptions. By taking the continuous limit of the discretization, a linear PDE is obtained. In another line of work begun by Kappen [9] the same linear PDE has been found through a particular transformation of the HJB. Existing approaches for solving the resulting linear HJB PDE have focused on the use of the Feynman-Kac Lemma, which relates the solution to the linear PDE to the diffusion of a stochastic process. This approach has been developed by Theodorou et al. [21] into the Path Integral framework in use with Dynamic Motion Primitives. Therein, sampling is augmented with the use of suboptimal policies, producing better estimates of the dynamics when executing an optimal policy. The resulting samples trajectories can then be used to in turn improve the policy, and then the process is iterated. These results have been developed in a number of compelling directions [23], [22], [3].

The sampling-based approach developed through the Feynman-Kac Lemma is an alternative to the approach presented here, with several potential advantages and disadvantages. Among these, sampling-based approaches such as that of Theodorou, may be more attractive in high dimensional state spaces. However, the approach presented in this paper may be quite rapid in practice, produces a global solution with no need for iteration, and the scalability of the process is an open question that this paper only begins to investigate. While it is beyond the scope of this paper, the framework presented below may find applications in the method of Control Lyapunov Functions [16] and Receding Horizon Control [7].

Matanya Horowitz and Joel Burdick are with the Department of Control and Dynamical Systems, Caltech, 1200 E California Blvd., Pasadena, CA. The corresponding author is available at mhorowit@caltech.edu. Matanya Horowitz is supported by a NSF Graduate Research Fellowship.

## II. BACKGROUND

Using techniques developed for polynomial optimization, we develop a method to obtain a universal approximation for the value function with the best error as measured in the pointwise norm. We will review the development of the linear optimal control PDE, along with the necessary tools of polynomial optimization.

### A. A Linear Hamilton-Jacobi-Bellman (HJB) Equation

A common construction in the optimization literature is the value function, which captures the “cost-to-go” from a given state. If such a quantity is known, the optimal action may be chosen as that which follows the gradient of the value, bringing the agent into the states with highest value over the remaining time horizon. The construction of the value function  $V(x)$  presented here follows the development in [24].

We focus on system with state  $x_t \in \mathbb{R}^n$  at time  $t$ , control input  $u_t \in \mathbb{R}^m$ , and dynamics that evolve according to the equation

$$dx_t = (f(x_t) + G(x_t)u_t) dt + B(x_t)\mathcal{L} d\omega_t \quad (1)$$

where the expressions  $f(x), G(x), B(x)$  are assumed to be polynomial functions of the state variables  $x$ , and  $\omega$  is a Brownian motion with (i.e., a stochastic process such that  $\omega_t$  has independent increments with  $\omega_t - \omega_s \sim N(0, t - s)$  for  $N(\mu, \sigma^2)$  a normal distribution). The matrix  $\mathcal{L}$  is constant. The system has costs  $r_t$  accrued at time  $t$  according to

$$r(x_t, u_t) = q(x_t) + \frac{1}{2}u_t^T R u_t \quad (2)$$

where  $q(x)$  is a state dependent cost and the control effort enters quadratically. We require  $q(x) \geq 0$  for all  $x$  in the problem domain (which is more carefully defined below).

The goal is to minimize the expected cost of the following functional,

$$J(x, u) = \phi_T(x_T) + \int_0^T r(x_t, u_t) dt \quad (3)$$

where  $\phi_T$  represents a state-dependent terminal cost. The solution to this minimization is known as the **value function**, where, beginning from an initial point  $x_t$  at time  $t$

$$V(x_t) = \min_{u_{t:T}} \mathbb{E}[J(x_t)] \quad (4)$$

with  $u_{t:T}$  being short-hand notation for  $u(t), t \in [t, T]$ .

The Hamilton-Jacobi-Bellman equation associated with this problem, arising from Dynamic Programming arguments [4], is found to be

$$-\partial_t V = \min_u \left( r + (\nabla_x V)^T f + \frac{1}{2} \text{Tr}((\nabla_{xx} V) G \Sigma_\epsilon G^T) \right) \quad (5)$$

where we define  $\Sigma_\epsilon = \mathcal{L}\mathcal{L}^T$ . As the control effort enters quadratically into the cost function it is a simple matter to solve for it analytically in (2):

$$u^* = -R^{-1}G^T \nabla_x V.$$

	Cost Functional	Desirability PDE
Finite	$\phi_T(x_T) + \int_0^T r(x_t, u_t) dt$	$\frac{1}{\lambda} q \Psi - \frac{\partial \Psi}{\partial t} = L(\Psi)$
First-Exit	$\phi_{T^*}(x_{T^*}) + \int_0^{T^*} r(x_t, u_t) dt$	$\frac{1}{\lambda} q \Psi = L(\Psi)$
Average	$\lim_{T \rightarrow \infty} \frac{1}{T} \mathbb{E} \left[ \int_0^T r(x_t, u_t) dt \right]$	$\frac{1}{\lambda} q \Psi - c \Psi = L(\Psi)$

TABLE I

CONSTRAINTS ON THE DESIRABILITY FUNCTION ARISING IN A NUMBER OF COMMON OPTIMAL CONTROL PROBLEMS [25].

The minimal control,  $u^*$ , may then be substituted into (5) to yield the following nonlinear, second order partial differential equation (PDE):

$$\begin{aligned} -\partial_t V &= q + (\nabla_x V)^T f - \frac{1}{2} (\nabla_x V)^T G R^{-1} G^T (\nabla_x V) \\ &+ \frac{1}{2} \text{Tr}((\nabla_{xx} V) B \Sigma_\epsilon B^T). \end{aligned}$$

The difficulty of solving this PDE is what usually prevents practitioners of optimal control from attempting to solve for the value function directly. However, it has recently been found [24], [25], [8] that with the assumption that there exists a  $\lambda \in \mathbb{R}$  and a control penalty cost  $R \in \mathbb{R}^{n \times n}$  satisfying this equation

$$\lambda G(x) R^{-1} G(x)^T = B(x) \Sigma_\epsilon B(x)^T \triangleq \Sigma_t \quad (6)$$

and using the logarithmic transformation

$$V = -\lambda \log \Psi \quad (7)$$

it is possible, after substitution and simplification, to obtain the following *linear* PDE from Equation (6).

$$-\partial_t \Psi = -\frac{1}{\lambda} q \Psi + f^T (\nabla_x \Psi) + \frac{1}{2} \text{Tr}((\nabla_{xx} \Psi) \Sigma_t). \quad (8)$$

This transformation of the value function, which we call here the *desirability* [25], provides an additional, computationally appealing method through which to calculate the value function.

Similar arguments may be made to develop value functions in an additional problems of interest. These are listed in Table I. For brevity, an expression common to the desirability equations is defined

$$L(\Psi) := f^T (\nabla_x \Psi) + \frac{1}{2} \text{Tr}((\nabla_{xx} \Psi) \Sigma_t) \quad (9)$$

*Remark 1:* The condition (6) can roughly be interpreted as a controllability-type condition: the system must have sufficient control to span (or counterbalance) the effects of input noise on the system dynamics. A degree of designer input is also given up, as the constraint restricts the design of the control penalty  $R$ , requiring that control effort be highly penalized in subspaces with little noise, and lightly penalized in those with high noise. Additional discussion is given in [25].

### B. Sum of Squares Programming

We provide a brief review on Sum of Squares (SOS) programming, with additional technical details available in [13]. These tools will be key in the development of approximate solutions to (8). In brief, (8) specifies a set of *partial differential* equality constraints that the optimal solution must satisfy. To develop instead a close approximation these equality constraints may be relaxed to inequalities. The optimization problem is then to find the best approximate solution that lies in the set of polynomials that satisfy these inequality constraints, known as a semialgebraic set. SOS provides a method to perform optimization over such a set.

Formally, a *semialgebraic set* is a subset of  $\mathbb{R}^n$  that is specified by a finite number of polynomial equations and inequalities. An example is

$$\{(x_1, x_2) \in \mathbb{R}^2 \mid x_1^2 + x_2^2 \leq 1, x_1^3 - x_2 \leq 0\}.$$

Such a set is not necessarily convex, and testing membership in the set is intractable in general [13]. As we will see, however, there exists a class of semialgebraic sets that are in fact semidefinite-representable. Key to this development will be first the ability to test for non-negativity of a polynomial.

A multivariate polynomial  $f(x)$  is a *sum of squares* (SOS) if there exist polynomials  $f_1(x), \dots, f_m(x)$  such that

$$f(x) = \sum_{i=1}^m f_i^2(x).$$

A seemingly unremarkable observation is that a sum of squares is always positive. Thus, a sufficient condition for non-negativity of a polynomial is that the polynomial is SOS. Perhaps less obvious is that membership in the set of SOS polynomials may be tested as a convex problem. We denote the function  $f(x)$  being SOS as  $f(x) \in \Sigma(x)$ .

**Theorem 2:** ([13]) Given a finite set of polynomials  $\{f_i\}_{i=0}^m \in \mathbb{R}^n$  the existence of  $\{a_i\}_{i=1}^m \in \mathbb{R}$  such that

$$f_0 + \sum_{i=1}^m a_i f_i \in \Sigma(x)$$

is a semidefinite programming feasibility problem.

Thus, while the problem of testing non-negativity of a polynomial is intractable in general, by constraining the feasible set to SOS the problem becomes tractable. The converse question, is a non-negative polynomial necessarily a sum of squares, is unfortunately false, indicating that this test is conservative [13]. Nonetheless, SOS feasibility will be sufficiently powerful for our purposes.

1) *The Positivstellensatz:* Using SOS theory, it is possible to determine whether a particular polynomial, possibly parameterized, is a sum of squares. The next step is to determine how to combine multiple polynomial inequalities, i.e. semialgebraic sets of the form

$$P = \{x \in \mathbb{R}^n \mid f_i(x) \geq 0 \text{ for all } i = 1, \dots, m\}$$

for polynomial functions  $f_i(x)$  where  $x \in \mathbb{R}^n$ . The answer is given by Stengle's *Positivstellensatz*.

**Theorem 3:** (Positivstellensatz [19]) The set

$$X = \{x \mid f_i(x) \geq 0, h_j(x) = 0 \text{ for all } i = 1, \dots, m, j = 1, \dots, p\}$$

is empty if and only if there exists  $t_i \in \mathbb{R}[x]$ ,  $s_i, r_{ij}, \dots \in \Sigma$  such that

$$-1 = s_0 + \sum_i h_i t_i + \sum_i s_i f_i + \sum_{i \neq j} r_{ij} f_i f_j + \dots$$

Although this theorem is presented in terms of feasibility, it is easily adapted for the purposes of optimization. Given the problem

$$\begin{aligned} \min & f_0(x) \\ \text{subject to} & f_i(x) \leq 0 \quad \forall i \in 1, \dots, k \end{aligned}$$

a slack factor  $\gamma$  may be introduced to frame the equivalent infeasibility problem

$$\begin{aligned} \max & \gamma \\ \text{s.t.} & \left. \begin{aligned} f_0(x) &\leq \gamma \\ f_i(x) &\leq 0 \quad \forall i \in 1, \dots, k \end{aligned} \right\} \text{infeasible} \end{aligned}$$

which is in a directly applicable form for the *Positivstellensatz*.

A sufficient condition for infeasibility may be created by limiting the inclusion of some of the multipliers, e.g. setting some to zero such as  $r_{ij}$  or  $r_{ijk}$ . Alternatively, it is possible to limit the degree of the multipliers  $h_i, s_i, r_{ij}$ . In the search for infeasibility we may therefore begin with a limited polynomial degree, increasing the degree if additional precision is required. This creates a *hierarchy* of semidefinite relaxations of increasing complexity but also with a decrease in the suboptimality of the solution. This construction is known more broadly as a Theta Body relaxation [5].

### III. SUM-OF-SQUARES RELAXATION OF THE HJB PDE

Sum of squares programming has found many uses in combinatorial optimization, control theory, and other applications. We now expand its use to include finding approximate solutions to the value function of the stochastic optimal control problem.

Obtaining solutions to linear PDEs is far from trivial. However, we propose to first approximate the desirability solution to the linear HJB PDE as a polynomial. While the value function may in fact be discontinuous, we make the modeling assumption that it may be approximated to a sufficiently high accuracy given a polynomial of sufficient degree. Furthermore, although the solution to the HJB is discontinuous in some locations, in many continuous domains, such as many robotics and control problems of interest, it will remain continuous over large portions of the domain. Historically, difficulties with the discontinuities present in HJB equations have led to the development of viscosity solutions [4], in effect placing a smoothness requirement on the solution.

We proceed with the finite horizon problem, but similar steps apply to all the problems listed in Table I. We make the assumption that the control problem occurs only on a

compact domain  $\mathbb{S}$  that is representable as a semialgebraic set, as is its boundary  $\partial\mathbb{S}$ .

The equality constraint of (8) may be relaxed, yielding the following constraints that are necessary for an over-approximation of the desirability function

$$\frac{1}{\lambda}q\Psi \leq \partial_t\Psi + f^T(\nabla_x\Psi) + \frac{1}{2}\text{Tr}((\nabla_{xx}\Psi)\Sigma_t) \quad (10)$$

Hereafter, we will indicate solutions to the above inequality as  $\Psi$ , and exact solutions to (8) as  $\Psi^*$ , the optimal desirability function. To obtain the best such approximation  $\Psi$  for a given polynomial order, the pointwise error of the approximation may be minimized in the optimization problem

$$\begin{aligned} \min \quad & \gamma \\ \text{s.t.} \quad & \gamma - \left( \frac{1}{\lambda}q\Psi - \partial_t\Psi - L(\Psi) \right) \geq 0 \end{aligned}$$

for  $x \in \mathbb{S}$ . The boundary conditions of (8) correspond to the exit conditions of the optimal control problem. In all problems this may correspond to colliding with an obstacle or goal region, and in the finite horizon problem there is the added boundary condition of the terminal cost at  $t = T$ . These final costs must then be transformed according to (7), producing the added constraint

$$\Psi|_{\partial\mathbb{S}} = e^{-\frac{\phi_T(x_T)}{\lambda}}$$

where  $\phi_T(x_T)$  is the terminal cost from (3). This constraint may be also be relaxed as an inequality. The complete optimization problem is then

$$\begin{aligned} \min \quad & \gamma \\ \text{s.t.} \quad & \frac{1}{\lambda}q\Psi \leq \partial_t\Psi + L(\Psi) \quad x \in \mathbb{S} \\ & \gamma \geq \frac{1}{\lambda}q\Psi - \partial_t\Psi - L(\Psi) \quad x \in \mathbb{S} \\ & \Psi \leq e^{-\frac{\phi_T(x)}{\lambda}} \quad x \in \partial\mathbb{S} \end{aligned} \quad (11)$$

As the inequalities are defined over polynomials, this optimization is defined over a semialgebraic set. This may be made tractible as follows.

*Proposition 4:* The optimization problem (11) where inequality constraints are relaxed to SOS membership may be solved as a semidefinite optimization program.

*Proof:* Let us propose a candidate solution to the optimization  $\Psi$ , a polynomial of fixed degree  $n$ , denoted  $\Psi_n$ . Each of the inequality constraints are non-negativity constraints over a polynomial and are therefore a semialgebraic set. The full set of constraints is an intersection of semialgebraic sets and therefore also a semialgebraic set. When the inequalities in this set are relaxed as SOS constraints, membership in the constraint set may be tested as a semidefinite program by Theorem 2. The optimization over this set is then enabled by Theorem 3. ■

Furthermore, one can in fact guarantee the exact and polynomial approximate desirability functions have a bounded relationship.

*Theorem 5:* Given a solution  $\{\Psi, \gamma\}$  to (11), and if  $\Psi^*$  is the solution to (8), then  $\Psi(x) \leq \Psi^*(x)$  for all  $x \in \mathbb{S}$ .

*Proof:* Consider the first-exit case for simplicity, and define the error between approximation  $\Psi$  and the optimal desirability  $\Psi^*$ ,  $e = \Psi - \Psi^*$ . Then, as all operators are linear,

$$\begin{aligned} \frac{1}{\lambda}qe &= \frac{1}{\lambda}q(\Psi - \Psi^*) \\ &= \frac{1}{\lambda}q\Psi - L(\Psi^*) \\ &\leq L(\Psi) - L(\Psi^*) \\ &\leq L(e) \end{aligned}$$

Defining the augmented operator  $P(e) := L(e) - \frac{1}{\lambda}qe$  then  $P$  is an elliptic operator and by the weak maximum principle for elliptic operators [18]

$$\sup_{\mathbb{S}} e \leq \sup_{\partial\mathbb{S}} e^+ \quad (12)$$

where  $e^+ = \max(e, 0)$  and  $e$  is non-positive on the boundary. Thus, the error remains less than zero everywhere, implying that  $\Psi \leq \Psi^*$ , and that  $\Psi$  is indeed a lower bound.

The weak maximum principle for parabolic operators can similarly be used in the case where the desirability PDE is parabolic. The only difference to note is that the augmented operator is now  $P(e) := L(e) + \partial_t - \frac{1}{\lambda}qe$ . For the weak maximum principle to be used, it is required that  $P(e)$  have the same form but with a negative temporal derivative  $P(e) = L(e) - \partial_t - \frac{1}{\lambda}qe$ . This is in fact the form of our operator, as the boundary condition along the time axis is assigned only at the terminal time, and the direction of time must be flipped in the proof relative to the time of the system's evolution. ■

*Remark 6:* This construction may be repeated for each of the objective functions found in Table I, albeit the average cost constant  $c$  must be determined a-priori in the average cost case [25].

Note that the principle underlying Proposition 4 may in fact be repeated with the inequalities reversed in optimization (11), resulting in a superharmonic error function. The result is that this reversed optimization is shown to be an over-approximation.

*Theorem 7:* The optimization problem (11) where inequality constraints are reversed and then replaced with SOS membership may be solved as a semidefinite program, and furthermore produces an upper bound  $\Psi$  of  $\Psi^*$  on the domain  $\mathbb{S}$ .

With upper and lower bounds to the optimal desirability function obtained, the distance between each and the optimal  $\Psi^*$  is bounded by a known value. Also, it is straightforward to relate these bounds to the value function as well.

*Proposition 8:* Given an upper (lower) bound  $\Psi$ , to a solution  $\Psi^*$  of (8), then  $V = -\lambda \log \Psi$  is a lower (upper) bound of  $V^*$ , the solution to (6).

*Proof:* For  $\Psi \geq \Psi^*$

$$\begin{aligned} V &= -\lambda \log \Psi \\ &\leq -\lambda \log \Psi^* \\ &= V^* \end{aligned}$$

Since  $\lambda$  is always positive. Similar reasoning applies to the lower bound. ■

*Remark 9:* Due to the nature of the log transformation (7),  $\Psi$  is necessarily positive on the domain  $\mathbb{S}$ . This may be included as an addition constraint  $\Psi \geq 0$  in (11). However, in this case the optimization for the lower bound of  $\Psi^*$  may not converge. It is possible to instead neglect this constraint and for its inapplicability to be remembered if the approximate desirability function  $\Psi$  is in fact less than zero at any point on the domain.

#### A. Analysis

Some preliminary analysis of this approach demonstrates several appealing qualities. The first of these is that the convergence of the algorithm is guaranteed.

*Proposition 10:* There exists a constant  $c$  such that the SOS optimization problem arising from (11) has a solution for all  $\gamma \geq c$

*Proof:* For the PDEs in Table I that are elliptic, all problem data is polynomial and therefore infinitely differentiable. By the elliptic regularity theorem, the solution  $\Psi$  is infinitely differentiable and therefore continuous. As this is a linear operator on a compact set, it is continuous if and only if it is bounded. Therefore there exists some constant  $c \geq \Psi$  on the domain  $\mathbb{S}$ . Similarly for the parabolic case the above holds true for each point in time, and integration of these finite quantities over a bounded time period also produces bounded solutions.

A constant polynomial  $p(x, t)$  may be taken to be the plane with  $p(x, t) = c$ . As this is a polynomial of degree zero, it is in the set of feasible solutions to (11). Since this is a convex problem, the existence of a feasible solution  $p(x, t)$  is sufficient for the algorithm to converge. ■

Intuitively, the previous result states that there must exist constant values that upper and lower bound the solution to the desirability, which are of course polynomial representable. Clearly such bounds may be quite poor in practice. However, placing this problem within a hierarchy of optimization problems with increasing polynomial degree we have the following result.

*Proposition 11:* Let  $\Psi_n$  be a polynomial approximation of the desirability function with maximum degree  $n$ . The hierarchy of SOS problems consisting of solutions to (11) with increasing polynomial degree produce a sequence of solutions  $\{\Psi_i, \gamma_i\}_{i \in I}$  with monotonically decreasing  $\gamma_i$

*Proof:* Clearly for a sufficiently high degree polynomial  $\Psi$ ,  $\Psi^*$  may be represented exactly if it is polynomial itself. Further, given a solution  $\Psi$  to (11), and an additional solution  $\Psi'$  of higher degree, each with solutions  $\gamma, \gamma'$  respectively,  $\gamma' \leq \gamma$  as  $\Psi'$  may achieve error  $\gamma$  by setting its additional degrees of freedom to zero, so the solution improves monotonically. ■

Note that we have no guarantee as to the divergence of the cost when executing the approximate value function from the true value function. We are only guaranteed that the value function is an over-approximation at a particular state. A consequence is illustrated in Figure 1. By following the gradient,

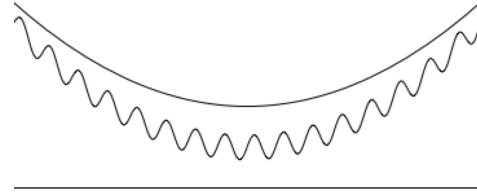


Fig. 1. Illustration of potential mis-alignment between value function gradients despite proximity of approximate value function. The x-axis here denotes state space domain, while the y-axis denotes the cost-to-go at a particular state.

$\deg(\Psi) \setminus \deg(s_i)$	2	4	6	8	10
2	1.0	1.0	1.0	1.0	0.9994
4	1.0	1.0	1.0	0.9999	0.9947
6	1.0	1.0	0.7508	0.7498	0.7406
8	1.0	1.0	0.2834	0.0592	0.0592
10	1.0	1.0	0.2834	0.0590	0.0487

TABLE II  
SOLUTION QUALITY  $\gamma$  OF THE DESIRABILITY LOWER BOUND FOR  
VARYING POLYNOMIAL DEGREE OF SOLUTION  $\Psi$  AND  
POSITIVESTELLENSATZ MULTIPLIERS  $s_i$ .

the system may diverge significantly from the optimal path, further undermining the accuracy of the approximate value function. This is an issue common to many approximate dynamic programming schemes [12], [2], [6]. A common technique employed is to simply use Monte Carlo simulation of the policy resulting from the approximate value solution, providing an upper bound  $J^{ub}$  on the realizable cost. Here, we may also flip the sign of the inequality (11) to also obtain a lower bound. If the resulting sampled upper bound  $J^{ub}$  is near this lower bound, then the policy may be said to be empirically near-optimal.

#### IV. EXAMPLES

A scalar and a two-dimensional pair of examples reveal preliminary results on the the computational characteristics of the method. In the following problems the optimization parser Yalmip [11] was used in conjunction with the semidefinite optimization package SDPT3 [26].

##### A. Scalar System Example

A nonlinear, unstable system with the following dynamics is considered

$$dx = (x^3 + 5x^2 + x + u) dt + d\omega \quad (13)$$

on the domain  $x \in \mathbb{S} = [-1, 1]$ . The problem chosen is a first-exit problem, with  $\phi(-1) = 10$ , and  $\phi(1) = 0$ . For this instance,  $\mathcal{L} = 1$ ,  $G = 1$ ,  $B = 1$ , and the cost parameters  $q = 1$ ,  $R = 1$  are assigned. Optimal solutions to (11) of the desirability for varying polynomial degree  $\deg(\Psi)$  are shown in Figure 2 along with its transformed cost-to-go. The pointwise error in the desirability for increasing polynomial degree on the solution and the multipliers is shown in Table II. The figures and the table clearly show that the higher the degree of polynomial approximation, the smaller the approximation error.

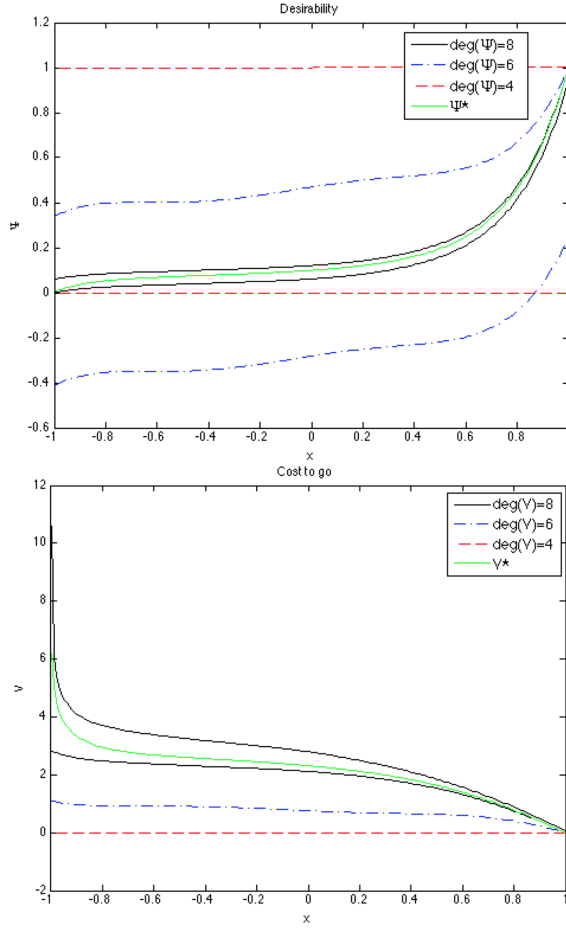


Fig. 2. Plots of approximate and exact desirability and cost-to-go solutions for scalar system (13) versus state  $x$ , in the interval  $x \in [-1, 1]$ . The dashed red, dashed blue, and solid black lines represent the  $\deg(\Psi) = 4$ ,  $\deg(\Psi) = 6$ , and  $\deg(\Psi) = 8$  approximations. The multipliers of the Positivstellensatz were set to have matching degree, i.e.  $\deg(s_i) = \deg(\Psi)$ .

### B. Two Dimensional Example

Next, a nonlinear 2-dimensional problem example adapted from [15] was solved as a first-exit problem. The dynamics are set as

$$\begin{bmatrix} dx \\ dy \end{bmatrix} = \left( \begin{bmatrix} -2x - x^3 - 5y - y^3 \\ 6x + x^3 - 3y - y^3 \end{bmatrix} + \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \right) dt + \begin{bmatrix} d\omega_1 \\ d\omega_2 \end{bmatrix}$$

The system was given the task of reaching a boundary of the domain  $\mathbb{S} = [-1, 1]^2$ , and once there would fulfill its task with no additional cost. The control penalty was set to  $R = I_{2 \times 2}$ , and state cost as  $q(x) = 0.1$ . The boundary conditions for the sides  $x = -1, y = 1, y = -1$  were set to have a penalty of  $\phi(x, y) = 1$ , while for the remaining boundary  $x = 1$  the boundary was set to have a quadratic cost  $\phi(x, y) = 1 - (y - 1)^2$ . The results are shown in Figure 3.

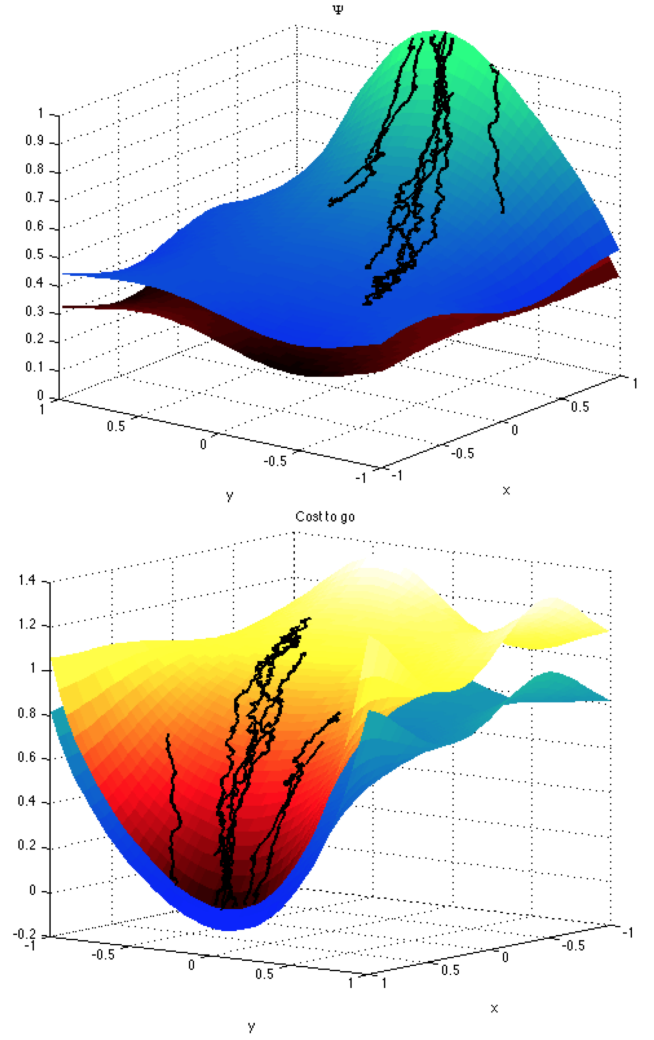


Fig. 3. Desirability and Value solutions for the two dimensional example. The problem was solved with  $\deg(\Psi) = \deg(s_i) = 14$ . The upper bound had gap  $\gamma_{up} = 0.0979$ , and lower bound  $\gamma_{lw} = 0.1049$ . Ten simulated trajectories of the closed loop system, randomly sampled from  $x, y \in [-.75, .75]^2$  are shown in black.

## V. DISCUSSION

A method to find the value function for a class of stochastic optimal control problems was proposed. Sum of squares and semidefinite programming was used to construct a global solution without recourse to value iteration or other forms of dynamic programming. The method produces a-priori bounds on the solutions' pointwise error from the optimal HJB solution. Unfortunately, a-priori error bounds on the cost of the trajectories resulting from policies which follow the approximate solution were not obtained, but are the subject of further investigation. As it stands, there is no guarantee that a specific objective will be obtained, e.g. to reach a goal region or provide stabilization. Indeed, the mis-alignment of true and approximate value functions has surfaced in the controls community [17] as well as in the broader literature on approximate dynamic programming [2].

The question remains of how the algorithms presented in

this paper differ from the simple process of applying approximate dynamic programming with polynomial basis functions. Key in this work is the development in the continuous state space of the problem. Although approximate dynamic programming aggregates states, it nonetheless begins from a discrete state space. The result is that the number of constraints in the corresponding dynamic program depends on the size of the discrete state space [2]. While in practice many of these constraints may be inactive, it isn't possible to determine a-priori the inactive ones. Furthermore, as has been shown, the SOS framework gives strong guarantees on the pointwise distance between the approximate and exact value functions.

There exists many interesting avenues for future investigation. Primary among these is the incorporation and analysis of systems whose dynamics are not polynomial functions of state and input. Although trigonometric functions were incorporated in several examples, a broader synthesis that does not require ad-hoc analysis, as well as one that could incorporate discontinuities, is needed.

Interesting connections exist with literature in the controls community. Therein, efforts have been made to use Lyapunov functions for optimal control, in this context dubbed Control Lyapunov Functions. Unfortunately, methods to produce optimal control Lyapunov functions have eluded researchers to date. The methods presented here seem promising in this light.

As mentioned, this method is proposed as an alternative to sampling based methods that utilize the Feynman-Kac lemma. A distinct advantage of the Feynman-Kac based approach is that the required sampling scales well with increasing dimension of the state space. It is an interesting question as to how the method proposed here can be extended to high dimensional state spaces. The expressivity of polynomials leads one to believe that they may be used to postpone the curse of dimensionality by not requiring as fine a partition as a straightforward MDP-like discretization for a given desired accuracy.

#### A. Acknowledgements

The authors would like to thank Venkat Chandrasakaran for guidance and suggestions. The first author is grateful for the support provided by a National Science Foundation graduate fellowship. This work was partially supported by DARPA, through the ARM-S and DRC programs, as well as the Robotics Technology Consortium Alliance (RCTA).

#### REFERENCES

- [1] D. P. Bertsekas. *Dynamic Programming and Optimal Control*. Athena Scientific, 1995.
- [2] D. P. de Farias and B. Van Roy. The linear programming approach to approximate dynamic programming. *Operations Research*, 51(6):850–865, 2003.
- [3] K. Dvijotham and E. Todorov. A unified theory of linearly solvable optimal control. *Artificial Intelligence (UAI)*, 2011.
- [4] W. H. Fleming and H. M. Soner. *Controlled Markov processes and viscosity solutions*, volume 25. Springer, New York, July 2006.
- [5] J. Gouveia, P. A. Parrilo, and R. R. Thomas. Theta bodies for polynomial ideals. *SIAM Journal on Optimization*, 20(4):2097–2118, 2010.
- [6] C. Guestrin, D. Koller, R. Parr, and S. Venkataraman. Efficient solution algorithms for factored MDPs. *J. Artif. Intell. Res. (JAIR)*, 19:399–468, 2003.
- [7] A. Jadbabaie, J. Yu, and J. Hauser. Unconstrained receding-horizon control of nonlinear systems. *Automatic Control, IEEE Transactions on*, 46(5):776–783, 2001.
- [8] H. Kappen. Linear Theory for Control of Nonlinear Stochastic Systems. *Physical Review Letters*, 95(20):200201, Nov. 2005.
- [9] H. J. Kappen. Path integrals and symmetry breaking for optimal control theory. *Journal of Statistical Mechanics: Theory and Experiment*, 2005(11):P11011–P11011, Nov. 2005.
- [10] S. M. LaValle. *Planning Algorithms* (2006). Cambridge Univ Press.
- [11] J. Lofberg. YALMIP : a toolbox for modeling and optimization in MATLAB. In *Computer Aided Control Systems Design, 2004 IEEE International Symposium on*, pages 284–289, 2004.
- [12] B. O'Donoghue. *Suboptimal Control Policies Via Convex Optimization*. PhD thesis, Stanford University, 2012.
- [13] P. A. Parrilo. Semidefinite programming relaxations for semialgebraic problems. *Mathematical Programming*, 96(2):293–320, May 2003.
- [14] P. A. Parrilo and S. Lall. Semidefinite programming relaxations and algebraic optimization in control. *European Journal of Control*, 9(2):307–321, 2003.
- [15] S. Prajna and A. Papachristodoulou. Analysis of switched and hybrid systems-beyond piecewise quadratic methods. 4:2779–2784, 2003.
- [16] J. Primbs. *Nonlinear optimal control: A receding horizon approach*. PhD thesis, California Institute of Technology, Apr. 1999.
- [17] J. A. Primbs, V. Nevistić, and J. C. Doyle. Nonlinear optimal control: A control Lyapunov function and receding horizon perspective. *Asian Journal of Control*, 1(1):14–24, 1999.
- [18] M. H. Protter and H. F. Weinberger. *Maximum Principles in Differential Equations*. Springer, 1984.
- [19] G. Stengle. A nullstellensatz and a positivstellensatz in semialgebraic geometry. *Mathematische Annalen*, 207(2):87–97, June 1974.
- [20] E. Theodorou. *Iterative path integral stochastic optimal control: Theory and applications to motor control*. PhD thesis, University of Southern California, June 2011.
- [21] E. Theodorou, J. Buchli, and S. Schaal. A generalized path integral control approach to reinforcement learning. *The Journal of Machine Learning Research*, 9999:3137–3181, 2010.
- [22] E. Theodorou, J. Buchli, F. Stulp, and S. Schaal. An Iterative Path Integral Reinforcement Learning Approach. In *Snowbird Learning Workshop*, July 2010.
- [23] E. A. Theodorou and E. Todorov. Relative entropy and free energy dualities: connections to path integral and kl control. pages 1466–1473, 2012.
- [24] S. F. B. J. Theodorou E and S. S. An Iterative Path Integral Stochastic Optimal Control Approach for Learning Robotic Tasks. pages 1–8, Apr. 2011.
- [25] E. Todorov. Efficient computation of optimal actions. *Proceedings of the National Academy of Sciences (PNAS)*, 106(28):11478–11483, 2009.
- [26] K.-C. Toh, M. J. Todd, and R. H. Tütüncü. SDPT3 – a MATLAB software package for semidefinite programming. *Optimization Methods and Software*, 11(1-4):545–581, 1999.